Distant Speech Recognition Experiments Using the AMI Corpus

Chapter 16
Distant Speech Recognition Experiments using the AMI Corpus

Steve Renals and Pawel Swietojanski

Abstract This chapter reviews distant speech recognition experimentation using the AMI Corpus of multiparty meetings. The chapter compares conventional approaches using microphone array beamforming followed by single-channel acoustic modelling with approaches which combine multichannel signal processing with acoustic modelling in the context of convolutional networks.

16.1 Introduction

Distant conversational speech recognition [30] poses many technical challenges such as multiple overlapping acoustic sources (including multiple talkers), reverberant acoustic environments, and highly conversational speaking styles. Microphone array-based approaches have been used to address the task since the early 1990s [29, 3, 20], and from about 2004 onwards there have been various evaluation frameworks for distant speech recognition including the multi-channel Wall Street Journal audio visual corpus (MC-WSJ-AV) [18], the NIST rich transcription (RT) series of evaluations [9], the REVERB challenge (chapter **??**), and the CHiME challenges (chapter **??**).

From 2004–2009, the NIST RT evaluations (`http://www.itl.nist.gov/iad/mig//tests/rt`) focused on the problem of meeting transcription, and enabled comparison between various automatic meeting transcription systems (e.g. [26, 14]). These evaluations of multiparty conversational speech recognition had a focus on meeting transcription. The acoustic data was classified by the recording condition: individual headset microphones (IHM), a single distant microphone (SDM), and multiple distant microphones (MDM). The MDM condition typically used tabletop microphone arrays, with the SDM condition choosing a single microphone from the array.

Centre for Speech Technology Research, University of Edinburgh, UK.
e-mail: `\{p.swietojanski,s.renals\}@ed.ac.uk`

For MDM systems, microphone array processing was usually distinct from speech recognition. For instance, the AMIDA MDM system of Hain et al [14] processed the multi-channel microphone array data using a Wiener noise filter, followed by weighted filter-sum beamforming based on time-delay-of-arrival (TDOA) estimates, postprocessed using a Viterbi smoother. In practice the beamformer tracked the direction of maximum energy, passing the beamformed signal onto a conventional ASR system – in the case of [14], a Gaussian mixture model / hidden Markov model (GMM/HMM) trained using the discriminative minimum phone error (MPE) criterion [21], speaker adaptive training [4], and the use of bottleneck features [12] derived from a neural network trained as a phone classifier. The resulting system employed a complex multi-pass decoding scheme, including substantial cross-adaptation and model combination.

One of the main principles underpinning "deep learning" is that systems for classification and regression can be constructed from multiple modules that are optimised using a common objective function [17]. In the context of distant speech recognition this can lead to approaches such as LIMABEAM [24, 25], in which the parameters of the microphone array beamformer are estimated so as to maximise the likelihood of the correct utterance model. Marino and Hain [19] explored removing the beamforming component entirely, and directly concatenating the feature vectors from the different microphones as the input features for a HMM/GMM speech recognition system. In contrast to the LIMABEAM approach which retains explicit beamforming parameters, but optimises them according to a criterion related to speech recognition accuracy, the concatenation approach makes the beamforming parameters implicit. More recently Xiao el al (Chapter **??**) introduced a neural network approach to optimise beamforming to maximise speech recognition performance, also allowing the beamforming and acoustic model to be optimised simultaneously, and Sainath et al introduced a multichannel neural network architecture operating on raw waveforms (Chapter **??**).

This chapter is concerned with distant speech recognition of meeting recordings, based on experiments employing the AMI Corpus (section 16.2). We present experiments using beamformed microphone array features as a baseline (section 16.3), comparing with systems using concatenated features from multiple channels (section 16.4), and systems using cross-channel convolutional networks (section 16.5).

16.2 Meeting corpora

Work on meeting transcription has been largely enabled by two corpora: the ICSI Meeting Corpus and the AMI Corpus. The ICSI Meeting Corpus (`http://www.icsi.berkeley.edu/Speech/mr/`) contains about 75 hours of recorded meetings with 3–15 participants, captured using individual headset microphones (IHM), as well as an MDM condition comprising 4 boundary microphones placed about 1m apart along the tabletop [15]. One limitation of this corpus was the fact that the distant microphones were widely spaced and not in known positions.

The AMI Corpus (http://corpus.amiproject.org) comprises over 100 hours of recordings of multiparty meetings. The meetings were recorded as part of the AMI/AMIDA projects (http://www.amiproject.org) using a common "Instrumented Meeting Room" (IMR) environment located at the University of Edinburgh, Idiap Research Institute, and TNO Human Factors (Figure 16.1). The corpus design, and the recording methodology, was driven by the multidisciplinary nature of the AMI/AMIDA projects, which included research in computer vision, multimodal processing, natural language processing, human-computer interaction, and social psychology, as well as speech recognition [6, 8]. The IMR recording environments each included at least six cameras (personal and room-view), multiple distant microphones (MDMs) configured as an eight-element circular microphone array placed on the meeting table, and a individual headset microphone (IHM) for each participant, as well as information capture using digital pens, smart whiteboards, shared laptop spaces, data projector, and videoconferencing if used. The different recorded streams were synchronised to a common timeline. In the initial recordings (2005) frame-level synchronisation was achieved using a hardware-based approach. Later meeting capture experiments used a high resolution spherical digital video camera system and a 20-element microphone array with software synchronisation, as well as further experiments using digital MEMS microphone arrays [31]. The corpus also contains a verbatim word-level transcription synchronised to the same timeline. Additional annotations include dialogue acts, topic segmentation, extractive and abstractive summaries, named entities, limited forms of head and hand gestures, gaze direction, movement around the room, and head pose information. NXT – the NITE XML Toolkit (http://groups.inf.ed.ac.uk/nxt/) – an XML-based open source software infrastructure for multimodal annotation [7] was used to carry out and manage the annotations.

About two-thirds of the AMI corpus consists of "scenario meetings" in which four participants play roles in a design across a set of four meetings, recorded in thirty replicas, ten in each of the IMRs. The remainder of the corpus comprises recordings of "real" meetings which would have taken place irrespective of the recording. The use of scenario meetings had several advantages in the context of the interdisciplinary nature of the projects in which the corpus was produced: it allowed preferred meeting outcomes to be designed into the process, allowing the definition of group outcome and productivity measures; the knowledge and motivation of the participants was controlled, thus removing the confounding factors that would be present in a set of real meetings (for example the history of relationships between the participants, and the organisational context); the meeting scenario could be replicated, enabling task-based evaluations. The main drawbacks of using scenario meetings are based around a reduction in diversity and naturalness. Although the recorded speech is spontaneous and conversational, the overall dialogue is less realistic. Furthermore, replicating the scenarios significantly reduces the linguistic variability across the corpus: for example in 100 hours of the AMI corpus there are about 8,000 unique words, about half the number observed in that duration in other corpora such as Wall Street Journal and Switchboard.

Fig. 16.1: AMI corpus recording setup.

16.3 Baseline speech recognition experiments

In this paper we focus on distant speech recognition using the AMI corpus. Unlike the NIST RT evaluations, where the AMI data was used together with other meeting corpora (e.g. [14, 12]), we have carefully defined the training, development, and test sets based on a 3-way partition of the AMI corpus, thus ensuring that our distant speech recognition experiments use identical microphone array configurations in the three different acoustic environments. The training, development, and test sets all include a mix of scenario- and non-scenario-based meetings, and are designed such that no speaker appears in more than one set. The definitions of these sets have also been made available on the AMI corpus website and are used in the associated Kaldi recipe (`https://github.com/kaldi-asr/kaldi/tree/master/egs/ami/`). We use the segmentation provided with the AMI corpus annotations (version 1.6.1). In this work, we consider all segments (including those with overlapping speech), and the speech recognition outputs are scored by the `asclite` tool [9] following the NIST RT recommendations for scoring simultaneous speech (`http://nist.gov/speech/tests/rt/2009`).

IHM recordings: Our baseline acoustic models used 13-dimension MFCCs (C0-C12), splicing together 7 frames, projecting down from 91 to 40 dimensions using linear discriminant analysis (LDA) [13], and decorrelated using a single semi-tied covariance (STC) transform [10] (also referred to as a maximum likelihood linear transform, MLLT). These features are referred to as LDA/STC. Both GMM-HMM and ANN-HMM acoustic models are speaker adaptively trained (SAT) on these LDA/STC features using a single CMLLR transform estimated per speaker. The GMM-HMM systems provide the state alignments for training the ANNs. Addi-

System	Microphone configurations				
	IHM	MDM8	MDM4	MDM2	SDM
AMI development set					
GMM BMMI on LDA/STC	30.2 (SAT)	54.8	56.5	58.0	62.3
ANN on LDA/STC	26.8 (SAT)	49.5	50.3	51.6	54.0
ANN on FBANK	26.8	49.2	-	50.1	53.1
AMI evaluation set					
GMM BMMI on LDA/STC	31.7 (SAT)	59.4	61.2	62.9	67.2
ANN on LDA/STC	28.1 (SAT)	52.4	52.6	52.8	59.0
ANN on FBANK	29.6	52.0	-	52.4	57.9

Table 16.1: Word error rates (%) for the GMM and ANN acoustic models for various microphone configurations.

tionally, we also train ANN systems on 40-dimension log Mel filterbank (FBANK) features appended with first and second temporal derivatives. The state alignments obtained using the LDA/STC features were used for training the ANNs on FBANK features.

SDM/MDM recordings: We used either a single element of the microphone array (SDM) or delay-sum beamforming on 2, 4, or 8 uniformly-spaced array channels[1] using the BeamformIt toolkit [5] (MDM); the audio is then processed in a similar fashion to the IHM configuration. The major difference between the IHM and SDM/MDM configurations is that when audio is captured with distant microphones, it is not realistically possible to ascribe a speech segment to a particular speaker without having performed speaker diarisation. Hence we do not use any form of speaker adaptation or adaptive training in the SDM/MDM experiments, unless stated otherwise.

For all acoustic conditions we trained: (1) GMM-HMM systems using LDA/STC features (speaker adapted in the IHM case) optimised according to the BMMI criterion; (2) ANN systems using LDA/STC features optimised according to the cross-entropy criterion; and (3) ANN systems using FBANK features optimised according to the cross-entropy criterion. We used a set of about 4,000 tied states in each configuration, with about 80.000 Gaussians in each GMM-based system. The GMM-based systems were used to provide the state alignments for training the corresponding ANNs. The ANN systems were feed-forward networks, each with 6 hidden layers of 2,048 units, employing sigmoid transfer functions. The baseline experimental results are summarised in Table 16.1.

While Table 16.1 presents the WER for all segments, including those with overlapped speech, Figure 16.2 shows the WERs for segments scored with different numbers of overlapped speakers. As expected, overlapped segments are harder to recognise. In fact, even if a beamformer can select the dominant source perfectly it still does not address the problem of recognising overlapped speech which would require source separation and independent decoding for each identified source. Fig-

[1] Mics 1 and 5 were used in the 2-mic case; mics 1, 3, 5, and 7 in the 4-mic case.

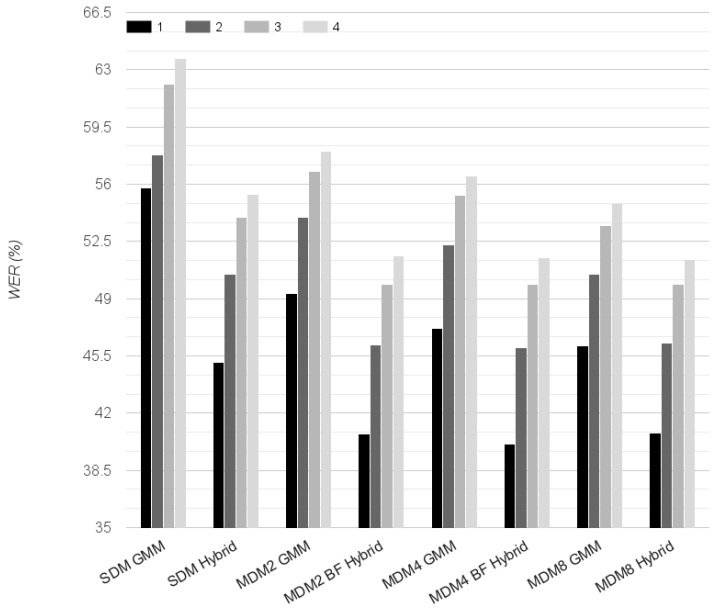

Fig. 16.2: Development set WERs for segments with 1, 2, 3 and 4 overlapping speakers. Acoustic models are trained on MFCC LDA/STC features. The Figure comes originally from [27] and the results are not directly comparable to the one reported in Table 16.1 due to the latter benefits from later refinements in the recipe. The figure was included to visualise the overlapping speakers issue across different systems.

ure 16.2 presents results for different systems in terms of the number of overlapping speakers in the segment. There is an 8–12% reduction in WER when only considering segments with non-overlapping speech. One can also notice that the WER deteriorates relatively more in the presence of overlapped speech for ANNs. For example, in the SDM case a 12% relative drop in WER is observed for the GMM-HMM and over 19% relative for the ANN-HMM system. This may be because ANNs model non-overlapped segments more accurately, and part of this advantage diminishes for fragments containing overlapping speech. We do not further address the issue of overlapping speakers in this chapter, and to keep the exposition simple we report WERs for all segments as they are (including overlapping speakers).

16.4 Channel concatenation experiments

As an alternative to beamforming it is possible to incorporate multiple channels in an ANN acoustic model by concatenating them, thus providing a sequence of higher

Table 16.2: WER for ANNs trained on multiple channels. SDM models are trained on channel 1.

Combining method	Recognition Channel(s)	AMI Dev Set
SDM (no combination)	1	53.1
SDM (no combination)	2	52.9
Concatenate 1+5	3,7	51.8
Concatenate 1+3+5+7	2,4,6,8	51.7
Multi-style 1+3+5+7	1	51.8
Multi-style 1+3+5+7	2	51.7

dimension acoustic vectors. We performed a set of experiments in order to evaluate the extent to which an ANN is able to learn to do front-end processing – both noise-cancellation and beamforming – by providing the features extracted from multiple microphones as input to the networks (cf [19]). In these experiments the networks again have 6 hidden layers,[2] with a wider input layer of concatenated channels. There are some differences to the baseline experiments, since Wiener filtering and beamforming are time domain operations, whereas the ANNs trained on concatenated features are operating entirely in either the cepstral or log-spectral domain. Nevertheless, the results offer an indication of the complementarity of the information from different channels. The results are tabulated in Table 16.2, and indicate that ANNs trained on concatenated inputs perform substantially better than the SDM case, achieving results approaching those obtained using beamforming. Since the ANNs trained on concatenated features do not use any knowledge of the array geometry, the technique is applicable to an arbitrary configuration of microphones.

To further understand the nature of the compensation being learned by the ANNs with multi-channel inputs, we performed an additional control experiment. The input to the ANN was from a single channel, and at test time this was identical to the SDM case. However, during training the data from other channels was also presented to the network, although not at the same time. In other words, the ANN is presented with data drawn from multiple channels while at test time it is only tested on a single channel. We call this the multi-style training, and it is related to our work on low-resource acoustic modelling [11], where a similar concept was used to train ANNs in a multilingual fashion. From Table 16.2 we see that this approach performs similarly to the ANNs with concatenated input, without requiring multiple channels at the recognition stage. Recognition results on channel 2, which is not used in the multi-style training, show similar trends. These results strongly suggest that there is information in a single channel to have more accurate recognition. However, extraneous factors in the data may confound a learner trained only on data from a single channel. Being forced to classify data from multiple channels using the same shared representation (i.e. the hidden layers) the network learns how to ignore the channel-specific covariates. To the best of our knowledge, this is the first

[2] However, since the networks are being tasked with additional processing, it may be that deeper architectures are more suitable.

result to show that it is possible to improve recognition of audio captured with a single distant microphone by guiding the training using data from microphones at other spatial locations.

16.5 Convolutional neural networks

A channel concatenation network may be enriched by constraining one or more of the lower layers to have local connectivity and to share parameters – a convolutional neural network (CNN). CNNs have defined the state of the art on many vision tasks [17] and can reduce the speech recognition word error rate (WER) when applied to acoustic modelling [2, 23]. The major conceptual difference between recent CNN structures for speech modelling and previous trials in the form of both CNNs [17] and the closely-related time-delay neural networks [16] lies in performing convolution and/or sharing parameters across frequency rather than time (see also Chapter **??**).

The input to a CNN comprises of FBANK features within an acoustic context window reordered such that each frequency band contain all the related static and dynamic coefficients. The hidden activations are then generated by a linear valid convolution[3] of a local frequency region. The same set of filters is then applied across different frequency regions to form a complete set of convolutional activations which can be subsampled, for instance by using the maxpooling operator, to further limit the variability across different frequencies.

Since the channels contain similar information (acoustic features shifted in time) we conjecture that the filter weights may be shared across different channels. Nevertheless, the formulation and implementation allow for different filter weights in each channel. Similarly, it is possible for each convolutional band to have a separate learnable bias parameter instead of the biases only being shared across bands [2, 23].

The complete set of convolutional layer activations is obtained by applying the (shared) set of filters across the whole (multi-channel) input space (as depicted in the top part of Fig 16.3). In this work the weights are tied across the input space; alternatively the weights may be partially shared, tying only those weights spanning neighbouring frequency bands. Although limited weight sharing was reported to bring improvements for phone classification [2] and small-scale tasks [1], a recent study on larger tasks [23] suggests that full weight sharing with a sufficient number of filters can work equally well, while being easier to implement.

Multi-channel convolution builds feature maps similarly to the LeNet-5 model [17] where each convolutional band is composed of filter activations spanning all input channels. We also constructed feature maps using max-pooling across channels, in which the activations are generated in channel-wise fashion and then max-pooled to

[3] The convolution of two vectors of size X and Y may result either in the vector of size $X+Y-1$ for a full convolution with zero-padding of non-overlapping regions, or the vector of size $X-Y+1$ for a valid convolution where only the points which overlap completely are considered.

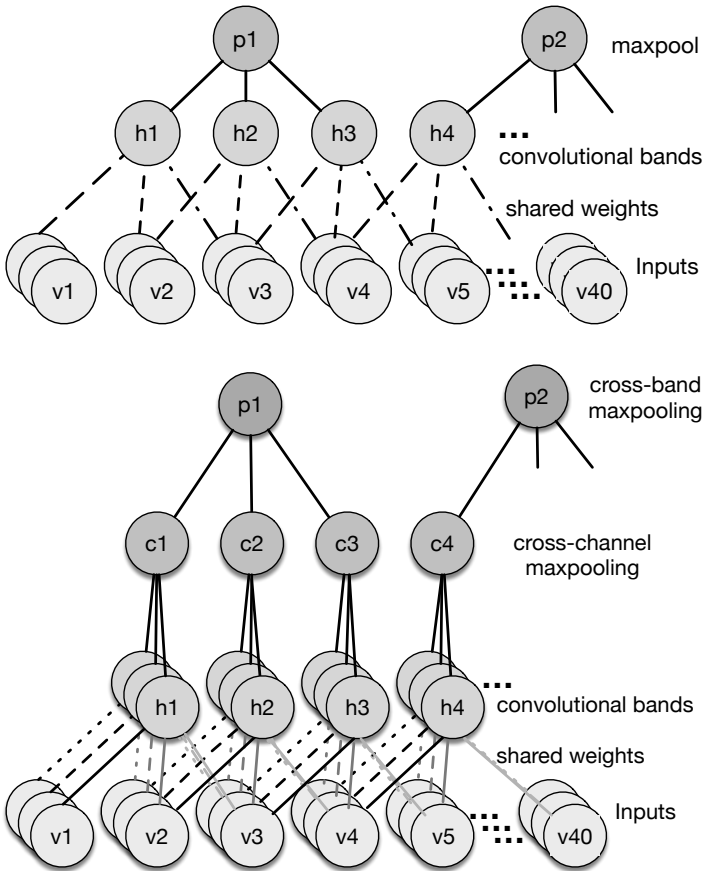

Fig. 16.3: Convolutional network layer with (top) cross-band maxpooling incorporating all channels, and (bottom) cross-channel maxpooling within each band, followed by cross-band maxpooling.

form a single cross-channel convolutional band. The resulting cross-channel activations can be further max pooled along frequency (Fig 16.3, bottom). Channel-wise convolution may be viewed as a special case of 2-dimensional convolution, where the effective pooling region is determined in frequency but varies in time depending on the actual time delays between the microphones. This CNN-based approach to multichannel speech recognition was first presented in [28, 22].

The CNN/ANN models in this section were trained on FBANK features appended with the first and the second time derivatives which were presented in an 11-frame window.

SDM recordings: The results of the single channel CNN can be found in Table 16.3 with the first two lines presenting the GMM and ANN baselines from Table 16.1. The following three lines are results for the CNN using max-pool sizes

Table 16.3: Word Error Rates (%) on AMI – SDM, where R is the pool size

System	AMI Dev Set
BMMI GMM-HMM (LDA/STC)	63.2
ANN (FBANK)	53.1
CNN ($R = 3$)	51.4
CNN ($R = 2$)	51.3
CNN ($R = 1$)	52.5

Table 16.4: Word Error Rates (%) on AMI – MDM.

System	AMI Dev Set
MDM with beamforming (8 microphones)	
BMMI GMM-HMM	54.8
ANN	49.5
CNN	46.8
MDM without beamforming	
ANN 4ch concatenated	51.2
CNN 2ch conventional	50.5
CNN 4ch conventional	50.4
CNN 2ch channel-wise	50.0
CNN 4ch channel-wise	49.4

of $R = N = 1,2,3$. By using CNNs we were able to obtain 3.4% relative reduction in WER with respect to the best ANN model and a 19% relative reduction in WER compared with a discriminatively trained GMM-HMM (baseline numbers taken from Table 16.1). The total number of parameters of the CNN models varies as $R = N$ while J is kept constant across the experiments. However, the best performing model had neither the highest nor the lowest number of parameters, which suggests it is due to the optimal pooling setting.

MDM recordings: For the MDM case we compared a delay-sum beamformer with the direct use of multiple microphone channels as input to the network. For the beamforming experiments, we follow noise cancellation using a Wiener filter with delay-sum beamforming on 8 uniformly-spaced array channels using the BeamformIt toolkit [5]. The results are summarised in Table 16.4. The first block of Table 16.4 presents the results for the case in which the models were trained on a beamformed signal from 8 microphones. The first two rows show the WER for the baseline GMM and ANN acoustic models as reported in Table 16.1. The following row contains the CNN model trained on 8 beamformed channels obtaining 2.7% absolute improvement (5.5% relative) over ANN. The configuration of the MDM CNN is the same as the best SDM CNN ($R = N = 2$).

The second part of a Table 16.4 shows WERs for the models directly utilising multi-channel features. The first row is a baseline ANN variant trained on 4 concatenated channels from Table 16.2. Then we present the CNN models with MDM input convolution performed as in Fig 16.3 (top) and a pooling size of 2, which was optimal for the SDM experiments. This scenario decreases WER by 1.6% relative

Table 16.5: Word Error Rates (%) on AMI Dev Set – IHM

System	WER(%)
BMMI GMM-HMM (SAT)	29.4
ANN	26.6
CNN	25.6

when compared to the ANN structure with concatenated channels (this approach can be seen as a channel concatenation for CNN models). Applying channel-wise convolution with two-way pooling (Fig 16.3, bottom) brings further gains of 3.5% WER relative. Furthermore, channel-wise pooling works better for more input channels: conventional convolution on 4 channels achieves 50.4% WER, practically the same as the 2 channel network, while channel-wise convolution with 4 channels achieves 49.5% WER, compared to 50.0% for the 2-channel case. These results indicate that picking the best information (selecting the feature receptors with maximum activations) within the channels is crucial when doing model-based combination of multiple microphones.

IHM recordings: We observe similar relative WER improvements between ANN and CNN for close talking speech recordings (Table 16.5) as were observed for the MDM and SDM experiments. The CNN achieves 3.6% WER reduction relative to the ANN model. Both ANN and CNN systems outperform a BMMI-GMM system trained in a speaker adaptive (SAT) fashion by 9.4% and 12.9% relative WER respectively. We did not see any improvements by increasing pooling size. Sainath et al [23] previously suggested that pooling may be task dependent.

16.6 Discussion and Conclusions

In this chapter we have presented some baseline experiments for distant speech recognition of multiparty meetings using the AMI corpus. ANN-based systems provide WER reductions compared with GMM-based systems, and further reductions are obtained using convolutional hidden layers followed by maxpooling. We present a number of experiments exploring the effect of replacing microphone array beamforming with ANN and CNN architectures that take multichannel input. Although multichannel CNNs do not out-perform beamforming approaches on the AMI corpus our results indicate that these CNN architectures are able to learn from multichannel signals. We have applied these approaches to the ICSI corpus, in which the microphone array is less calibrated, and our results indicated that cross-channel CNN architectures out-perform beamforming by a small amount [22].

Our current experiments do not explicitly attempt to optimise the acoustic model for overlapping talkers, or for reverberation. The promising results using raw multiple channel input features in place of beamforming open the possibilities to learning representations taking into account aspects such as overlapping speech. One inter-

esting research direction is the use raw waveform features in a multichannel context, as discussed in Chapter **??**.

References

1. Abdel-Hamid, O., Deng, L., Yu, D.: Exploring convolutional neural network structures and optimization techniques for speech recognition. In: Proc. ICSA Interspeech (2013)
2. Abdel-Hamid, O., Mohamed, A.R., Hui, J., Penn, G.: Applying convolutional neural networks concepts to hybrid NN–HMM model for speech recognition. In: Proc. IEEE ICASSP, pp. 4277–4280 (2012)
3. Adcock, J., Gotoh, Y., Mashao, D., Silverman, H.: Microphone-array speech recognition via incremental MAP training. In: Proc IEEE ICASSP, pp. 897–900 (1996)
4. Anastasakos, T., McDonough, J., Schwartz, R., Makhoul, J.: A compact model for speaker-adaptive training. In: Proc ICSLP, pp. 1137–1140 (1996)
5. Anguera, X., Wooters, C., Hernando, J.: Acoustic beamforming for speaker diarization of meetings. IEEE Trans. Audio, Speech, & Lang. Process. **15**, 2011–2021 (2007)
6. Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska, A., McCowan, I., Post, W., Reidsma, D., Wellner, P.: The AMI meeting corpus: A pre-announcement. In: Processings Machine Learning for Multimodal Interaction (MLMI), pp. 28–39 (2005)
7. Carletta, J., Evert, S., Heid, U., Kilgour, J.: The NITE XML toolkit: Data model and query language. Language Resources and Evaluation **39**, 313–334 (2005)
8. Carletta, J., Lincoln, M.: Data collection. In: S. Renals, H. Bourlard, J. Carletta, A. Popescu-Belis (eds.) Multimodal Signal Processing: Human Interactions in Meetings, chap. 2, pp. 11–27. Cambridge University Press (2012)
9. Fiscus, J., Ajot, J., Radde, N., Laprun, C.: Multiple dimension Levenshtein edit distance calculations for evaluating ASR systems during simultaneous speech. In: Proc LREC (2006)
10. Gales, M.: Semi-tied covariance matrices for hidden Markov models. IEEE Transactions on Speech and Audio Processing **7**(3), 272–281 (1999)
11. Ghoshal, A., Swietojanski, P., Renals, S.: Multilingual training of deep neural networks. In: Proc. IEEE ICASSP (2013)
12. Grezl, F., Karafiat, M., Kontar, S., Cernocky, J.: Probabilistic and bottle-neck features for LVCSR of meetings. In: Proc. IEEE ICASSP, pp. IV–757–IV–760 (2007)
13. Haeb-Umbach, R., Ney, H.: Linear discriminant analysis for improved large vocabulary continuous speech recognition. In: Proc. IEEE ICASSP, pp. 13–16 (1992). URL `http://dl.acm.org/citation.cfm?id=1895550.1895555`
14. Hain, T., Burget, L., Dines, J., Garner, P., Grézl, F., El Hannani, A., Karafíat, M., Lincoln, M., Wan, V.: Transcribing meetings with the AMIDA systems. IEEE Transactions on Audio, Speech and Language Processing **20**, 486–498 (2012)
15. Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., Wooters, C.: The ICSI meeting corpus. In: Proc IEEE ICASSP, pp. I–364–I–367 (2003)
16. Lang, K., Waibel, A., Hinton, G.: A time-delay neural network architecture for isolated word recognition. Neural Networks **3**, 23–43 (1990)
17. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc IEEE **86**, 2278–2324 (1998)
18. Lincoln, M., McCowan, I., Vepa, J., Maganti, H.: The multi-channel Wall Street Journal audio visual corpus (MC-WSJ-AV): Specification and initial experiments. In: Proc IEEE ASRU (2005)
19. Marino, D., Hain, T.: An analysis of automatic speech recognition with multiple microphones. In: Proc Interspeech, pp. 1281–1284 (2011)

20. Omologo, M., Matassoni, M., Svaizer, P., Giuliani, D.: Microphone array based speech recognition with different talker-array positions. In: Proc IEEE ICASSP, pp. 227–230 (1997)
21. Povey, D., Woodland, P.: Minimum phone error and I-smoothing for improved discriminative training. In: Proc IEEE ICASSP, pp. 105–108 (2002)
22. Renals, S., Swietojanski, P.: Neural networks for distant speech recognition. In: Proc HSCMA (2014)
23. Sainath, T., Kingsbury, B., Mohamed, A., Dahl, G., Saon, G., Soltau, H., Beran, T., Aravkin, A., Ramabhadran, B.: Improvements to deep convolutional neural networks for LVCSR. In: Proc IEEE ASRU (2013)
24. Seltzer, M., Raj, B., Stern, R.: Likelihood-maximizing beamforming for robust hands-free speech recognition. IEEE Trans. Speech, & Audio Process. **12**, 489–498 (2004)
25. Seltzer, M., Stern, R.: Subband likelihood-maximizing beamforming for speech recognition in reverberant environments. IEEE Trans. Audio, Speech, & Lang. Process. **14**, 2109–2121 (2006)
26. Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Janin, A., Magimai-Doss, M., Wooters, C., Zheng, J.: The SRI-ICSI Spring 2007 meeting and lecture recognition system. In: R. Stiefelhagen, R. Bowers, J. Fiscus (eds.) Multimodal Technologies for Perception of Humans, no. 4625 in LNCS, pp. 373–389. Springer (2008)
27. Swietojanski, P., Ghoshal, A., Renals, S.: Hybrid acoustic models for distant and multichannel large vocabulary speech recognition. In: Proc. IEEE ASRU (2013). DOI 10.1109/ASRU.2013.6707744
28. Swietojanski, P., Ghoshal, A., Renals, S.: Convolutional neural networks for distant speech recognition. IEEE Signal Processing Letters **21**, 1120–1124 (2014)
29. Van Compernolle, D., Ma, W., Xie, F., Van Diest, M.: Speech recognition in noisy environments with the aid of microphone arrays. Speech Commun. **9**, 433–442 (1990)
30. Wölfel, M., McDonough, J.: Distant Speech Recognition. Wiley (2009)
31. Zwyssig, E., Lincoln, M., Renals, S.: A digital microphone array for distant speech recognition. In: Proc. IEEE ICASSP, pp. 5106–5109 (2010). DOI 10.1109/ICASSP.2010.5495040